A comparison of graph clustering algorithms
Jean Creusefond

To cite this version:
Jean Creusefond. A comparison of graph clustering algorithms. International Symposium on Web AlGorithms, Jun 2015, Deauville, France. hal-01171341

HAL Id: hal-01171341
https://hal.archives-ouvertes.fr/hal-01171341
Submitted on 3 Jul 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
A comparison of graph clustering algorithms

Jean Creusefond
GREYC, Normandy University
jean.creusefond@unicaen.fr

I. INTRODUCTION

The community detection problem is very natural: given a set of people and their relationships, can we understand the underlying structure of social groups? The applications are numerous in marketing, politics, social statistics, . . .

Thanks to technological improvements and change of uses, many social networks of various sizes has been automatically extracted from recorded social relationships. The most obvious examples are social network websites, but other networks are also studied as social networks: collaboration between scientists, who-talks-to-whom on the phone/on emails, etc. Automatic extraction made large networks available for study, and the community detection algorithms that we could evaluate with ease by watching the result on small instances before, can not be compared on real-world networks.

We therefore try to find common ground among the various clustering algorithms. Indeed, most of them share design similarities, as the underlying assumptions about the characteristics of communities or the general steps of the algorithm. Our experiments show to what extent these similarities imply similarities of results.

Our work is close to the one of Almeida et al. [1], that compared a good number of algorithms and a few quality functions. However, they did not directly compare clusterings, but only the result when a quality function is applied on them.

II. GRAPHS AND ALGORITHMS

We want to run available algorithms on all graphs, we select relatively small real-world graphs (< 100K edges) to stay in reasonable computation times.

- Karate [12], relationships in a karate club;
- Dolphins [8], a community of dolphins;
- Football [6], matches between football teams;
- Netscience [10], scientific co-authorships;
- Facebook [7], friends in a facebook excerpt.

In this preliminary work, we compare 4 algorithms:

- Clauset [3]: an optimisation for sparse graphs of the hierarchical modularity optimisation algorithm proposed by Newman [9], where communities are recursively merged if it corresponds to to merge improving the global modularity the most;
- Leading eigenvector [10]: a spectral method based on the modularity matrix, the eigenvector of the largest positive eigenvalue is used to split the network in two. This method is applied recursively;
- Louvain [2]: at first, vertex are communities. Each node then changes community affiliation to the one of its neighbour if it improves modularity. The process is repeated with communities as vertex until no improvement can be made;
- Conclude [5]: a greedy algorithm using k-path centrality and the Louvain Method.

III. COMPARING CLUSTERINGS

The Normalised Mutual Information [4] (NMI) is used to compare two clusterings (see Eq. 1). I(A, B) is the mutual information of the two sets (an information-theoric measure of the the dependancy), that is normalised by the sum of the entropy (H(·)) of both clusterings.

IV. QUALITY FUNCTIONS

Modularity [6]:

$$Q(C) = \sum_{c \in C} \left[ \frac{E(c)}{m} - \left( \frac{Vol(c)}{2m} \right)^2 \right]$$ (2)

Mean internal clustering coefficient:

$$Cl(C) = \sum_{v \in V} \frac{|\{a, b\} \in k_C^C(v), \{a, b\} \in E|}{\left( k_C^C(v) \right)^2}$$ (3)

With $k_C^C$ the set of neighbors of $v$ that are in the same community.

V. RESULTS

Do algorithms output similar results?

The average and standard deviation of the NMI are presented in the following table:

<table>
<thead>
<tr>
<th></th>
<th>conclude</th>
<th>clause</th>
<th>leading</th>
</tr>
</thead>
<tbody>
<tr>
<td>louvain</td>
<td>0.87±0.05</td>
<td>0.83±0.07</td>
<td>0.64±0.17</td>
</tr>
<tr>
<td>conclude</td>
<td>1.0±0.0</td>
<td>0.77±0.10</td>
<td>0.59±0.18</td>
</tr>
<tr>
<td>clause</td>
<td>1.0±0.0</td>
<td>0.64±0.18</td>
<td>1.0±0.0</td>
</tr>
</tbody>
</table>

Louvain and Conclude are very similar, and that this trend does not particularly depend on the dataset (since the standard variation is low). It is surprising, since they make different assumptions about the characteristics of a community. They do, however, share the same agglomerative strategy.

Clauset is quite close to both of these methods, while having a different agglomerative strategy. But they all share some kind of greedy process explaining the closeness of these methods. On the other hand, leading eigenvector, due to its completely different approach of the cluster formation, produces results that are less similar to the others.
Do quality functions behave accordingly?

We rank the quality of each algorithm for each quality measure and each graph. The ranking process is done on a larger benchmark, with a total of ten algorithms. The global ranking is given here, to reflect the distance between algorithms. The first value is the modularity rank, while the other one is the mean internal clustering coefficient rank.

<table>
<thead>
<tr>
<th></th>
<th>louvain</th>
<th>conclude</th>
<th>clausten</th>
<th>leading</th>
</tr>
</thead>
<tbody>
<tr>
<td>karate</td>
<td>1/1</td>
<td>5/2</td>
<td>6/8</td>
<td>4/9</td>
</tr>
<tr>
<td>dolphins</td>
<td>1/5</td>
<td>9/2</td>
<td>5/6</td>
<td>6/4</td>
</tr>
<tr>
<td>football</td>
<td>1/8</td>
<td>7/6</td>
<td>8/9</td>
<td>9/10</td>
</tr>
<tr>
<td>netscience</td>
<td>1/8</td>
<td>9/7</td>
<td>3/1</td>
<td>6/4</td>
</tr>
<tr>
<td>facebook</td>
<td>1/4</td>
<td>8/2</td>
<td>7/3</td>
<td>6/7</td>
</tr>
</tbody>
</table>

Surprisingly, Louvain and Conclude show almost symetrically opposed results. Indeed, Louvain is systematically the best method w.r.t modularity, while having a relatively bad internal clustering coefficient. On the other hand, Conclude is very good with the clustering coefficient and in the low end of modularity. It means that, while having clustering that share a lot of common ground (one clustering can be easily recomposed from the other), they have different characteristics, measured by the quality functions. Clauset does an average performance, rarely diverging from around the middle of the rankings, and so does the leading eigenvector method.

VI. FUTURE WORKS

We assume that, if a clustering algorithm give close results to another one, it is because they agree on underlying models. Quality functions can help to define what models are exactly used, by characterizing what a ”good” clustering is and measuring which algorithm agrees with this characterisation.

But quality functions themselves often capture the same notion : inside/outside density ratio, difference with a null model, etc. Which means that they often are redundant. We intend to compare clusterings that rank high w.r.t. different quality functions, and to discover relationships between these measures.

Once a few central quality function are identified, we can check their compliance to a set of axioms such as in [11]. The non-compliance of one of these axioms may imply a counter-intuitive behaviour, and should warn the user of the possible bias.

REFERENCES


